



DEVENIR UN ATHLETE

DE HAUT NIVEAU DU CODE

# Qui suis-je ?

Je suis Christelle, une développeuse Java depuis plus de 20 ans et crafteuse depuis quelques années.

 [www.linkedin.com/in/christelle-prut](https://www.linkedin.com/in/christelle-prut)

 [github.com/lifelightdev](https://github.com/lifelightdev)

 <https://lifelightdev.blogspot.com/>

Quel est le point commun entre  
un athlète de haut niveau  
comme Teddy Riner  
et un développeur ?

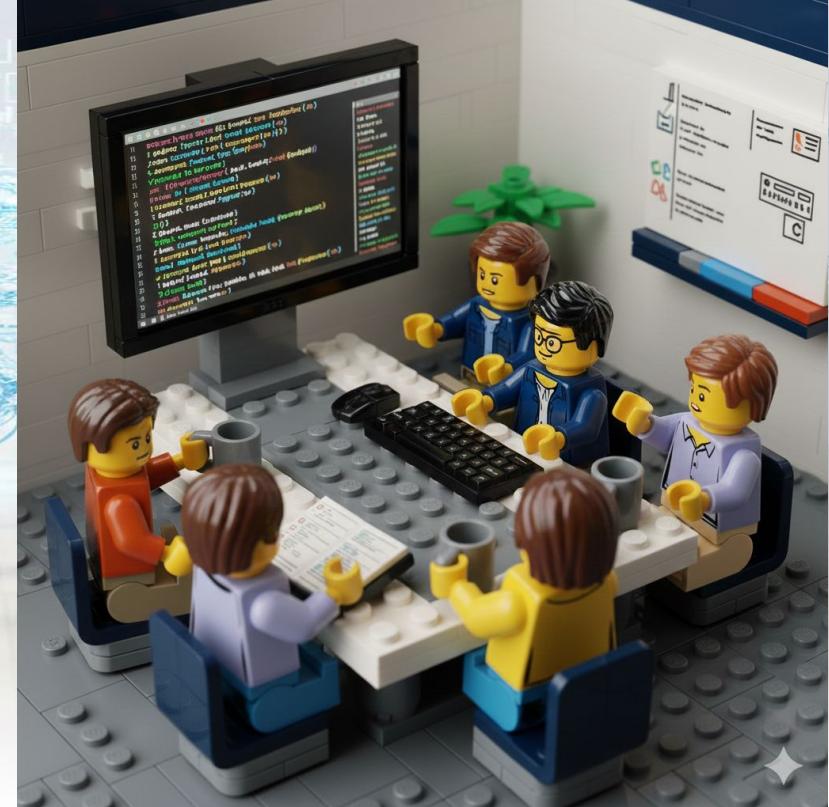




Un indice ?



# L'entraînement



# Un peu d'histoire



# Quelques définitions

**Kata** : C'est un petit exercice de programmation que l'on peut répéter à l'infini pour s'améliorer.

**Dojo** : C'est l'espace sécurisé dans lequel on pratique un kata

**Le craft** : c'est un artisan logiciel qui suis les principes du manifeste (Logiciel bien conçu, l'ajout constant de valeur, une communauté de professionnels)

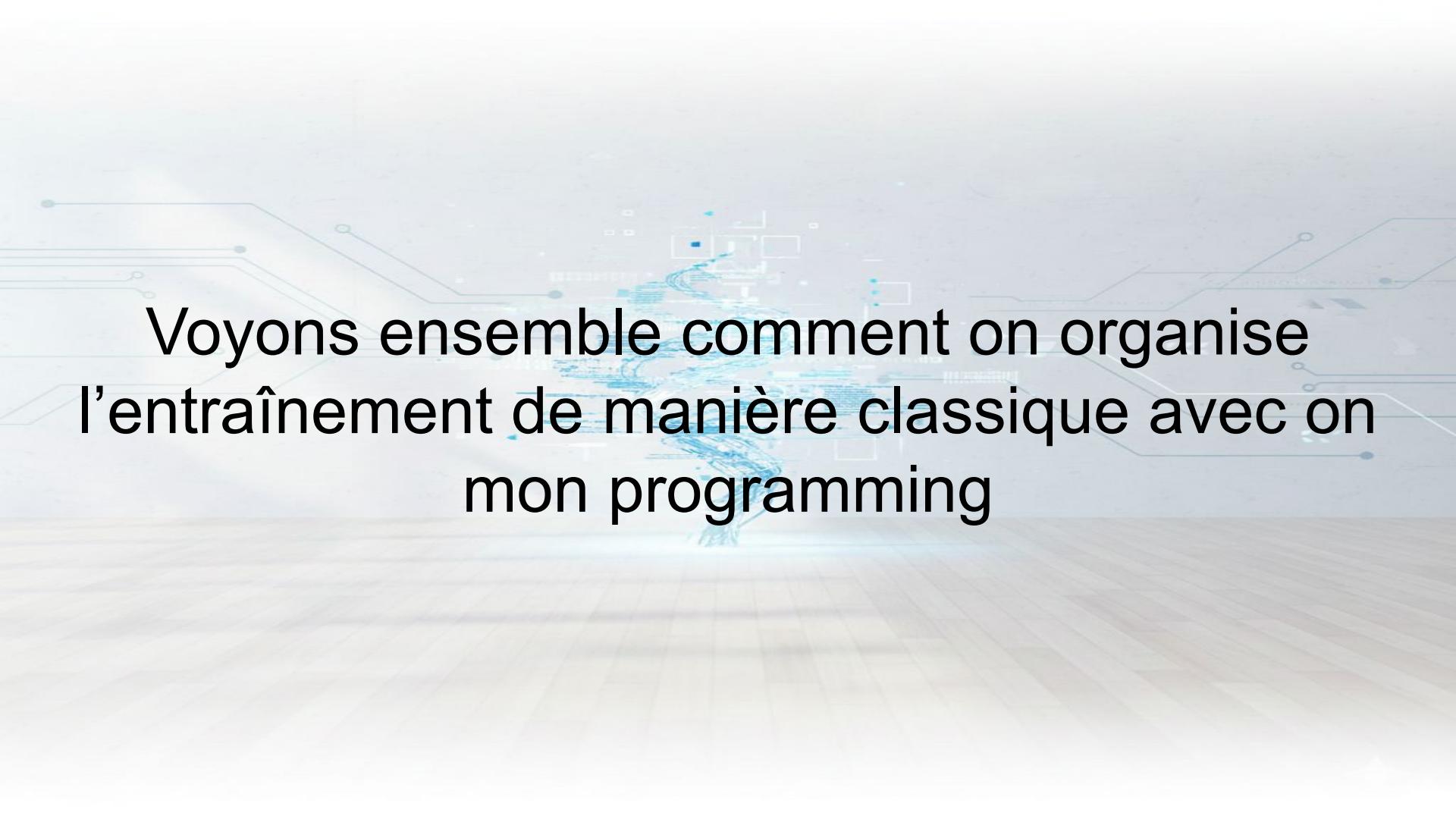
# Définissons un cadre pour apprendre en toute sérénité

- Zéro pression.

- Pas de pression sur le résultat
- Le niveau est sans importance.
- Code éphémère
- Toutes les questions sont bonnes à être posées.

- Respect mutuel.

- Esprit positif
- Pas de jugement.
- Le droit à l'erreur.



Voyons ensemble comment on organise  
l'entraînement de manière classique avec on  
mon programming

Il y a trois rôles :

- **L'assemblée** : elle représente l'ensemble des participants. Et, elle décide collectivement *quoi* coder. C'est comme un cerveau.
- **Le navigateur** : Il est l'intermédiaire entre l'Assemblée et le driver. Il dit *ce que le driver doit coder*. C'est comme la voix.
- **Le driver** : Il est au clavier. Il écrit ce que le navigateur lui demande sans prendre d'initiative sur le code. C'est les mains.

Voici quelques variantes, par exemple :

- **Le Randori** : Toutes les personnes alternent entre les rôles à intervalle fixe.
- **Le Fishbowl** : C'est un mécanisme de rotation libre et spontanée (semblable aux chaises musicales).

# Les contraintes

Pour rendre les exercices différents, il est possible d'ajouter des contraintes, par exemple :

- Faire du **TDD** (Test Driven Development) .
- Faire du **TCR** (Test Commit Revert).
- Faire du **TCRDD**, c'est la combinaison du TDD et du TCR.
- Faire du **TCRDD Gamble**, c'est du TCRDD avec un pari.

# Les astuces pour maximiser un bon moment pour tous

- Faire un rappel du fonctionnement d'un dojo pour les nouveaux et à la demande.
- Avoir un environnement de développement prêt à l'utilisation (pour ne pas passer le temps du dojo à installer l'environnement).
- Avoir une personne qui connaît bien le langage (pour ne pas passer le temps du dojo à aller lire la documentation du langage).
- Avoir un affichage lisible pour tous.
- Si la durée le permet, faire une orientation au milieu du dojo. Cela permet d'obtenir un feedback pour améliorer la fin de la session.
- Faire une rétrospective en fin de session.

# Lien utile

- Timer pour faire tourner les rôles du mob : <https://mobtime.hadrienmp.fr/>
- Système de vote : <https://emmanuelaatz.com/dojosurvey>
- Site de kata traditionnel : <https://codingdojo.org/kata/>
- Dojo en présentiel à Paris :  
[https://mobilizon.fr/@dojo\\_de\\_programmation\\_paris/events](https://mobilizon.fr/@dojo_de_programmation_paris/events)
- Dojo en distanciel avec les crafteurs de Lyon :  
<https://www.meetup.com/fr-fr/software-craftsmanship-lyon/>
- Dojo en distanciel avec les crafteurs de Lille :  
<https://www.meetup.com/software-craftsmanship-lille/>
- Le manifeste du Craft : <https://manifesto.softwarecraftsmanship.org/#/fr-fr>

# Conclusion

Pour **moi**, faire des kata en coding dojo cela me permet :

- D'expérimenter et de faire des découvertes
- D'apprendre et d'acquérir de nouvelles compétences.

# À vous la parole.

